✓

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/680,953	10/07/2003	Christopher John Stakutis	SJO920030049US1	6058

46917 7590 11/15/2006

KONRAD RAYNES & VICTOR, LLP.
ATTN: IBM37
315 SOUTH BEVERLY DRIVE, SUITE 210
BEVERLY HILLS, CA 90212

EXAMINER	
KIM, PAUL	
ART UNIT	**PAPER NUMBER**
2161	

DATE MAILED: 11/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/680,953	**Applicant(s)** STAKUTIS ET AL.
	Examiner Paul Kim	**Art Unit** 2161

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *24 August 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-35* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-35* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is responsive to the following communication: Amendment filed on 24 August 2006.

2. Claims 1-35 are pending and present for examination.

Response to Amendment

3. Claims 1, 14, 20, and 23 have been amended.

4. No claims have been cancelled.

5. No claims have been added.

Specification

6. As per the objections to the Specification, applicant's amendment has been acknowledged. Accordingly, the objections have been withdrawn.

Claim Rejections - 35 USC § 112

7. As per the rejections under 35 U.S.C. 112, applicant's amendment has been acknowledged. Accordingly, the rejections have been withdrawn.

Claim Rejections - 35 USC § 101

8. As per the rejections under 35 U.S.C. 101, applicant's amendment has been acknowledged. Accordingly, the rejections have been withdrawn.

Claim Rejections - 35 USC § 103

9. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. **Claim 1, 2, 10-11, 13-15, 18, 20-21, 23-24, 32-33, and 35** are rejected under 35 U.S.C. 103(a) as being unpatentable over Gonos (U.S. Patent No. 6,901,418, hereinafter referred to as GONOS), filed on 7 May 2002, and issued on 31 May 2005, in view of Crouse et al (U.S. Patent No. 5,764,972, hereinafter referred to as CROUSE), filed on 7 June 1995, and issued on 9 June 1998, and in view of Hug et al (U.S. Patent No. 5,806,078, hereinafter referred to as HUG), filed on 9 June 1994, and issued on 8 September 1998.

GONOS, CROUSE, and HUG disclose the limitations of claims 1, 2, 10-11, 13-15, 18, 20-21, 23-24, 32-33, and 35 for the reasons stated below.

GONOS differs from the claimed invention in that GONOS fails to expressly and specifically disclose a method of (1) determining whether a retention rule is provided for the source file in response to the write request, (2) in response to determining that one retention rule is provided for the source file, generating a versioned file name, wherein a versioned file comprises the source file at a point-in-time, and (3) processing the retention index file to determine whether to purge versioned files according to the retention rule provided for the source file (claims 1, 10, 14, 18, 20, 23 and 32).

GONOS differs from the claimed invention in that GONOS fails to expressly and specifically disclose a method of (1) determining versioned files to purge according to the retention rule and (2) transmitting a command to the file system to delete versioned files having the determined versioned file names (claims 2, 15, 21 and 24).

GONOS differs from the claimed invention in that GONOS fails to expressly and specifically disclose a method wherein retention index files are maintained at local storage to the host system and accessed locally by the host system to determine versioned files to purge according to retention rules (claims 11 and 33).

GONOS differs from the claimed invention in that GONOS fails to expressly and specifically disclose a method wherein the retention rule identifies a name of one source file to which the rule applies or identifies a source that created the source file to which the rule applies (claims 13 and 35).

11. **As per independent claims 1 and 23**, GONOS, in combination with CROUSE and HUG, discloses:

A method, comprising:

processing a request to write to a source file in a storage system {See GONOS, col. 5, lines 37-40, wherein this reads over "a process controls a processor to archive data and the structure used for the archived data. The process is initiated when a determination is made that a particular dataset is going to be archived"};

determining whether a retention rule is provided for the source file in response to the write request {See CROUSE, col. 20, lines 40-64, wherein this reads over "the RM module examines the direct access parameter of the file migration attributes to determine if direct access to the remote file is allowed"; and col. 21, lines 43-46, wherein this reads over "[i]f all selection criteria specified in the selection attributes are met by the file attributes and the file access information, that that remote file is eligible for archiving and/or removal"};

in response to determining that one retention rule is provided for the source file, generating a versioned file name, wherein a versioned file comprises the source file at a point-in-time {See HUG, col. 6, lines 28-41, wherein this reads over "identification data preferably identifies the date and time the version was created, a version name associated with the version"};

transmitting a command to a file system to copy the source file data to a versioned file {See GONOS, col. 6, lines 25-34, wherein this reads over "processor copies to the archive file the rows that meet the archiving criteria"} having the generated versioned file name {See CROUSE, col. 21, lines 3-16, wherein this reads over "making an archival copy of that file . . . up to four different copies of a backup/archive image of a remove file can be created"};

adding the generated versioned file name to a retention index file {See GONOS, col. 6, lines 43-50, wherein this reads over "[t]he archive index file and the archive index identification file describe the data stored in the archive file for the data set"}; and

processing the retention index file to determine whether to purge versioned files according to the retention rule provided for the source file {See CROUSE, col. 21, lines 24-28, wherein this reads over "the AR module automatically begins to purge or archive remote files that are eligible for elimination or archiving in accordance with their hierarchically selectable archival file attributes"}.

The combination of inventions disclosed in GONOS, CROUSE, and HUG would disclose a method comprising of processing a request to write to a source file in a storage system (i.e. a process that controls a processor to archive data"), determining whether a retention rule is provided (i.e. through a selection criteria of file attributes), generating a version name (i.e. creating a version name associated with the version), transmitting a command to copy the source file data to a version filed (i.e. copying rows that meet the archive criteria to an archive file), adding the versioned file name to a retention index file (i.e. the archive index file describe the data stored in the archive file), and processing the index file to determine whether to purge the version files (i.e. accessing the index file to see whether or not the archive files are eligible for elimination or archiving). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, and GONOS.

GONOS, CROUSE, and HUG are analogous art because they are from the same field of endeavor of data archival and recovery. One of ordinary skill in the art would have been motivated to do this modification so that copies of different versions of a file may be retained according to a retention rule in response to a request to write th file in a storage system.

12. **As per dependent claims 2, 15, 21, and 24**, GONOS, in combination with CROUSE and HUG, discloses:

> The method of claim 1, wherein purging the versioned files comprises:
>
> determining versioned files to purge according to the retention rule {See CROUSE, col. 21, lines 24-28, wherein this reads over "the AR module automatically beings to purge or archive remote files that are eligible for elimination or archiving in accordance with their hierarchically selectable archival file attributes"};
>
> deleting the determined versioned file names from the retention index file {See GONOS, col. 7, lines 60-67, wherein this reads over "When processing of each database table in the dataset is complete (step 580), the processor may delete unnecessary archive information (step 590). For example, the processor may delete the archive file, the archive index file, and the archive index identification file after a predetermined period of time has passed since those files were created or a predetermined period of time has passed since those file were used."}; and
>
> transmitting a command to the file system to delete versioned files having the determined versioned file names {See CROUSE, col. 21, lines 24-28, wherein this reads over "the AR module automatically begins to purge or archive remote files that are eligible for elimination"}.

The combination of inventions disclosed in CROUSE, HUG, and GONOS would disclose a method wherein purging the versioned file comprises of determining which versioned files to purge (i.e. determining which files are eligible for elimination or archiving), deleting the versioned file names from the retention index file (i.e. the processor may delete unnecessary archive information), and transmitting a command to delete the versioned files (i.e. the AR module automatically begins to purge or archive remote files that are eligible for elimination). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, and GONOS.

GONOS, CROUSE, and HUG are analogous art because they are from the same field of endeavor of data archival and recovery. One of ordinary skill in the art would have been motivated to do this modification so that purge-eligible versioned files may be deleted from the system and the versioned filed names may be removed from the index file.

13. **As per dependent claims 10, 18 and 32,** GONOS, in combination with CROUSE and HUG, discloses:

The method of claim 1,

wherein the operations of

processing the request {See CROUSE, col. 20, lines 40-64, wherein this reads over "in response to a request to read a remove file that is presently stored on a removable media"},

determining whether one retention rule is provided {See CROUSE, col. 20, lines 40-64, wherein this reads over "the RM module examines the direct access parameter of the file migration attributes to determine if direct access to the remote file is allowed"; and col. 21, lines 43-46, wherein this reads over "[i]f all selection criteria specified in the selection attributes are met by the file attributes and the file access information, that that remote file is eligible for archiving and/or removal"},

generating a new versioned file name {See HUG, col. 6, lines 28-41, wherein this reads over "identification data preferably identifies the date and time the version was created, a version name associated with the version"},

transmitting the command {See CROUSE, col. 21, lines 3-16, wherein this reads over "making an archival copy of that file . . . up to four different copies of a backup/archive image of a remove file can be created"},

adding the generated versioned file name to the retention index file {See GONOS, col. 6, lines 43-50, wherein this reads over "[t]he archive index file and the archive index identification file describe the data stored in the archive file for the data set"} and

processing the retention index file are performed by a host system {See CROUSE, col. 21, lines 24-28, wherein this reads over "the AR module automatically begins to purge or archive remote files that are eligible for elimination or archiving in accordance with their hierarchically selectable archival file attributes"}and

> wherein the versioned file, source file, and file system are on a remote storage system {See CROUSE, col. 7, line 65 – col. 8, line 2, wherein this reads over "the remote files stored on the remote secondary storage system and the local files stored on the local secondary storage system are stored and accessed in accordance with a file system tree structure"}.

The combination of inventions disclosed in CROUSE, HUG, and GONOS would disclose a method comprising of processing a request to write to a source file in a storage system (i.e. a process that controls a processor to archive data"), determining whether a retention rule is provided (i.e. through a selection criteria of file attributes), generating a version name (i.e. creating a version name associated with the version), transmitting a command to copy the source file data to a version filed (i.e. copying rows that meet the archive criteria to an archive file), adding the versioned file name to a retention index file (i.e. the archive index file describe the data stored in the archive file), and processing the index file to determine whether to purge the version files (i.e. accessing the index file to see whether or not the archive files are eligible for elimination or archiving). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, and GONOS.

GONOS, CROUSE, and HUG are analogous art because they are from the same field of endeavor of data archival and recovery. One of ordinary skill in the art would have been motivated to do this modification so that copies of different versions of a file may be maintained.

14. **As per dependent claims 11 and 33**, GONOS, in combination with CROUSE and HUG, discloses:

> The method of 10, wherein retention index files are maintained at local storage to the host system and accessed locally by the host system to determine versioned files to purge according to retention rules {See CROUSE, col. 7, line 65 – col. 8, line 2, wherein this reads over "the remote files stored on the remote secondary storage system and the local files stored on the local secondary storage system are stored and accessed in accordance with a file system tree structure"}.

The combination of inventions disclosed in CROUSE, HUG, and GONOS would disclose a method wherein retention index files are maintained at local storage to the host system and accessed locally by the host system. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, and GONOS.

GONOS, CROUSE, and HUG are analogous art because they are from the same field of endeavor of data archival and recovery. One of ordinary skill in the art would have been motivated to do this modification so that the retention index files may be readily accessible to the host system for determining which versioned filed to purge according to the retention rules.

15. **As per dependent claims 13 and 35**, GONOS, in combination with CROUSE and HUG, discloses:

> The method of claim 1, wherein the retention rule identifies a name of one source file to which the rule applies or identifies a source that created the source file to which the rule applies {See CROUSE, col. 20, lines 40-64, wherein this reads over "the RM module examines the direct access parameter of the file migration attributes to determine if direct access to the remote file is allowed"; and col. 21, lines 43-46, wherein this reads over "[i]f all selection criteria specified in the selection attributes are met by the file attributes and the file access information, that that remote file is eligible for archiving and/or removal"}.

The combination of inventions disclosed in CROUSE, HUG, and GONOS would disclose a method wherein the retention rule identifies a name of a source file or source to which the rule applies (i.e. application of the selection criteria to the files in order to determine which are eligible for archiving and/or removal). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, and GONOS.

GONOS, CROUSE, and HUG are analogous art because they are from the same field of endeavor of data archival and recovery. One of ordinary skill in the art would have been motivated to do this modification so that the rules may be properly assigned and applied to certain source files.

16. **As per independent claims 14 and 20**, GONOS, in combination with CROUSE and HUG, discloses:

> A system, comprising:
>
> a first storage system including source files {See CROUSE, col. 7, line 65 – col. 8, line 2, wherein this reads over "the remote files stored on the remote secondary storage system and the local files stored on the local secondary storage system are stored and accessed in accordance with a file system tree structure"} ;
>
> a second storage system including retention rules for the source files {See CROUSE, Figure 9; and col. 17, lines 56-67, wherein this reads over "four program modules that are called . . . the removable media manager (RM) module, and the archiving (AR) module, the file system (FS) module"};

a file system {See CROUSE, col. 4, lines 22-25, wherein this reads over "an archiving file system"};

means for processing a request to write to one of the source files in the first storage system {See CROUSE, col. 20, lines 40-64, wherein this reads over "in response to a request to read a remove file that is presently stored on a removable media"};

means for determining whether one of the retention rules is provided for the source file in response to the write request {See CROUSE, col. 20, lines 40-64, wherein this reads over "the RM module examines the direct access parameter of the file migration attributes to determine if direct access to the remote file is allowed"; and col. 21, lines 43-46, wherein this reads over "[i]f all selection criteria specified in the selection attributes are met by the file attributes and the file access information, that that remote file is eligible for archiving and/or removal"};

means for generating a versioned file name in response to determining that one retention rule is provided for the source file, wherein a versioned file comprises the source file at a point-in-time {See HUG, col. 6, lines 28-41, wherein this reads over "identification data preferably identifies the date and time the version was created, a version name associated with the version"};

transmitting a command to the file system to copy the source file data to a versioned file having the generated versioned file name in the first storage system {See CROUSE, col. 21, lines 3-16, wherein this reads over "making an archival copy of that file . . . up to four different copies of a backup/archive image of a remove file can be created"};

means for adding the generated versioned file name to a retention index file in the second storage system {See GONOS, col. 6, lines 43-50, wherein this reads over "[t]he archive index file and the archive index identification file describe the data stored in the archive file for the data set"}; and

means for processing the retention index file to determine whether to purge versioned files according to the retention rule provided for the source file {See CROUSE, col. 21, lines 24-28, wherein this reads over "the AR module automatically begins to purge or archive remote files that are eligible for elimination or archiving in accordance with their hierarchically selectable archival file attributes"}.

The combination of inventions disclosed in CROUSE, HUG, and GONOS would disclose a method comprising of processing a request to write to a source file in a storage system (i.e. a process that controls a processor to archive data"), determining whether a retention rule is provided (i.e. through a selection criteria of file attributes), generating a version name (i.e. creating a version name associated with the version), transmitting a command to copy the source file data to a version filed (i.e. copying rows that meet the archive criteria to an archive file), adding the versioned file name to a retention index file (i.e. the archive index file describe the data stored in the archive file), and processing the index file to determine whether to purge the version files (i.e. accessing the index file to see whether or not the archive files are eligible for elimination or archiving). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, and GONOS.

GONOS, CROUSE, and HUG are analogous art because they are from the same field of endeavor of data archival and recovery. One of ordinary skill in the art would have been motivated to do this modification so that copies of different versions of a file may be maintained.

17. **Claims 3-4, 16, 22, and 25-26** are rejected under 35 U.S.C. 103(a) as being unpatentable over CROUSE, in view of HUG and GONOS, and in further view of Howard (U.S. Patent No. 6,098,079, hereinafter referred to as HOWARD, filed on 2 April 1998, and issued on 1 August 2000.

CROUSE, HUG, and GONOS teach the limitations of claims 1, 2, 10-11, 13-15, 18, 20-21, 23-24, 32-33, AND 35 for the reasons stated above.

CROUSE, HUG, and GONOS differ from the claimed invention in that they fail to expressly and specifically disclose a method wherein the generated versioned file name is added to the retention index (claims 3, 16, 22, and 25).

CROUSE, HUG, and GONOS differ from the claimed invention in that they fail to expressly and specifically disclose a method wherein the function comprises a hash function (claims 4 and 26).

18. **As per dependent claims 3, 16, 22, and 25,** GONOS, in combination with CROUSE, HUG and HOWARD, discloses:

The method of claim 1, further comprising:

applying a function to the source file name to determine the retention index file {See GONOS, col. 6, lines 45-46, wherein this reads over "[a]n archive index file and an archive index identification file are associated with each archive file"},

wherein the determined retention index file maintains names of versioned files for the source file to which the function is applied {See HOWARD, col. 2, lines 48-52, wherein this reads over "[t]he reconciliation technique uses a set of journal files in which the history of file creation, modification, and deletion throughout the system is recorded"}, and

wherein the generated versioned file name is added to the determined retention index {See HOWARD, col. 2, lines 48-52, wherein this reads over "[t]he reconciliation technique uses a set of journal files in which the history of file creation, modification, and deletion throughout the system is recorded"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and HOWARD would disclose a method wherein a function is applied to the source file to determine the retention index file (i.e. the index file is associated with each archive file), and wherein the retention index file maintains the versioned files (i.e. a set of journal files which record the history of creation, modification, and deletion of files). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and HOWARD.

GONOS, CROUSE, HUG and HOWARD are analogous art because they are from the same field of endeavor of data archival and file versioning. One of ordinary skill in the art would have been motivated to do this modification so that the versioned file names may be added to the retention index.

19. **As per dependent claims 4 and 26**, GONOS, in combination with CROUSE, HUG and HOWARD, discloses:

The method of claim 3,

> wherein the function comprises a hash function {See HOWARD, col. 3, lines 31-40, wherein this reads over "[e]ach version entry contains a hash code . . . [that] uniquely identifies the contents of a corresponding version of the file"}, and
>
> wherein one retention index may maintain versioned file names for multiple source files to which the function is applied {See HOWARD, col. 2, lines 48-52, wherein this reads over "[t]he reconciliation technique uses a set of journal files in which the history of file creation, modification, and deletion throughout the system is recorded"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and HOWARD would disclose a method wherein the function comprises a hash function and the retention index may maintain versioned file names for multiple source files. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and HOWARD.

GONOS, CROUSE, HUG and HOWARD are analogous art because they are from the same field of endeavor of data archival and file versioning. One of ordinary skill in the art would have been motivated to do this modification in order to create a unique retention index file name, allowing for one retention index file to maintain information on versioned file names for different source files.

20. **Claims 5-9, 17, and 27-31** are rejected under 35 U.S.C. 103(a) as being unpatentable over CROUSE, in view of HUG and GONOS, and in further view of Kaler et al (U.S. Patent No. 6,928,447, hereinafter referred to as KALER), filed on 21 May 2004, and issued on 9 August 2005.

GONOS, CROUSE and HUG teach the limitations of claims 1, 2, 10-11, 13-15, 18, 20-21, 23-24, 32-33, AND 35 for the reasons stated above.

GONOS, CROUSE and HUG differ from the claimed invention in that they fail to disclose a method wherein the versioned file names for the source file are ordered on a timestamp (claims 5, 17, and 27).

21. **As per dependent claims 5, 17, and 27**, GONOS, in combination with CROUSE, HUG and KALER, discloses:

> The method of claim 1, wherein processing the retention index file to determine whether to purge versioned files according to the retention rule further comprises:
>
> > sorting the versioned file names for the source file in the retention index file ordered on a timestamp included in the versioned file names {See KALER, Figure 23}; and
> >
> > selecting versioned files from the sorted versioned file names to purge {See CROUSE, col. 21, lines 43-46, wherein this reads over "[i]f all selection criteria specified in the selection attributes are met by the file attributes and the file access information then that remote file is eligible for archiving and/or removal"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the versioned file names are sorted according to their timestamps, and accordingly selected for purge (i.e. purged on the basis of the recorded timestamp). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and KALER.

GONOS, CROUSE, HUG, and KALER are analogous art because they are from the same field of endeavor of data tracking and file versioning. One of ordinary skill in the art would have been motivated to do this modification so that the versioned files may be deleted according to the sort order.

22. **As per dependent claims 6 and 28**, GONOS, in combination with CROUSE, HUG and KALER, discloses:

> The method of claim 5, wherein the retention rule specifies a maximum number of versioned files to have for the source file {See CROUSE, col. 15, lines 19-49, wherein this reads over "[u]p to four copies of archival media may be specified for each remote file. Whether a remote file will be archived, how many copies will exist . . . is determined by the following parameters in the file migration attribute"}, and wherein selecting versioned files to purge comprises:
>
> > determining whether a number of the sorted versioned file names exceeds the maximum number {See CROUSE, col. 15, lines 1-16, wherein this reads over "[w]hen a new version of a file is created the previous version will be saved as a cycle if cycles are enabled for that file. The user may specify the number of cycles to be maintained and their life space by setting the following attributes" and "Cycle Limit"}; and
> >
> > selecting a number of oldest sorted versioned file names to purge to reduce the number of versioned file names in the retention index file to reach the maximum number {See CROUSE, col. 21, lines 28-30, wherein this reads over "[t]he remote files that are eligible for removal or archiving and have waited longest since last access will be eliminated or archived first"; and GONOS, col. 7, lines 60-67, wherein this reads over "the processor may delete the archive file, the archive index file, and the archive index identification file after a predetermined period of time has passed since those files were created or a predetermined period of time has passed"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the retention rule specifies a maximum number of versioned files to have for the source file (i.e. how many copies of a source file may exist), and selecting the versioned files to purge comprises of determining whether the number of version file names exceeds a maximum number (i.e. cycle limit), and selecting the oldest versioned file names for purge (i.e. archiving or removing those files which have waited longest since last access or after a predetermined period of time has passed). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and KALER.

GONOS, CROUSE, HUG, and KALER are analogous art because they are from the same field of endeavor of data tracking and file versioning. One of ordinary skill in the art would have been motivated to do this modification so that older versioned files are systematically and routinely purged from the system.

23. **As per dependent claims 7 and 29**, GONOS, in combination with CROUSE, HUG and KALER, discloses:

> The method of claim 5, wherein the retention rule specifies a time limit on versioned files to maintain for the source file, and wherein selecting versioned files to purge comprises:

> determining whether sorted version file names exceed the time limit according to the timestamp for the sorted versioned file names {See CROUSE, col. 15, lines 1-16, wherein this reads over "[w]hen a new version of a file is created the previous version will be saved as a cycle if cycles are enabled for that file. The user may specify the number of cycles to be maintained and their life space by setting the following attributes" and "Cycle Life Span"}; and
>
> selecting versioned file names in the retention index whose timestamp exceeds the time limit to purge versioned files whose timestamp exceeds the time limit {See CROUSE, col. 21, lines 28-30, wherein this reads over "[t]he remote files that are eligible for removal or archiving and have waited longest since last access will be eliminated or archived first"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the retention rule specifies a time limit on versioned files to be maintained for the source file (i.e. how long a versioned file of a source file may exist), and selecting the versioned files to purge comprises of determining whether the number of version file names exceeds a time limit (i.e. cycle life span), and selecting the versioned file names which exceed the time limit for purge (i.e. archiving or removing those files which have waited longest since last access or after a predetermined period of time has passed). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and KALER.

GONOS, CROUSE, HUG, and KALER are analogous art because they are from the same field of endeavor of data tracking and file versioning. One of ordinary skill in the art would have been motivated to do this modification so that older versioned files are systematically and routinely purged from the system.

24. **As per dependent claims 8 and 30**, GONOS, in combination with CROUSE, HUG and KALER, discloses:

> The method of claim 5, wherein the retention rule specifies a time period and maximum number of versioned files to maintain for the source file within the specified time period, and wherein selecting versioned files to purge comprises:
>
> determining whether a number of the sorted versioned file names within the specified time period exceeds the specified maximum number {See CROUSE, col. 15, lines 1-16, wherein this reads over "[w]hen a new version of a file is created the previous version will be saved as a cycle if cycles are enabled for that file. The user may specify the number of cycles to be maintained and their life space by setting the following attributes", "Cycle Limit", and "Cycle Life Span"}; and

> selecting a number of oldest sorted versioned file names to purge to reduce the number of versioned file names in the retention index file in the specified time period to reach the maximum number {See CROUSE, col. 21, lines 28-30, wherein this reads over "[t]he remote files that are eligible for removal or archiving and have waited longest since last access will be eliminated or archived first"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the retention rule specifies a time limit on versioned files to be maintained for the source file (i.e. how long a versioned file of a source file may exist), and selecting the versioned files to purge comprises of determining whether the number of version file names exceeds a time limit (i.e. cycle life span), and selecting the versioned file names which exceed the time limit for purge (i.e. archiving or removing those files which have waited longest since last access or after a predetermined period of time has passed). Furthermore, the combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the retention rule specifies a maximum number of versioned files to have for the source file (i.e. how many copies of a source file may exist), and selecting the versioned files to purge comprises of determining whether the number of version file names exceeds a maximum number (i.e. cycle limit), and selecting the oldest versioned file names for purge (i.e. archiving or removing those files which have waited longest since last access or after a predetermined period of time has passed). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and KALER.

GONOS, CROUSE, HUG, and KALER are analogous art because they are from the same field of endeavor of data tracking and file versioning. One of ordinary skill in the art would have been motivated to do this modification so that older versioned files are systematically and routinely purged from the system.

25. **As per dependent claims 9 and 31,** GONOS, in combination with CROUSE, HUG and KALER, discloses:

The method of claim 8,

> wherein the retention rule specifies multiple time periods and one maximum number for each time period to separately maintain versioned files for the source file for

different time periods {See CROUSE, col. 15, lines 1-16, wherein this reads over "[w]hen a new version of a file is created the previous version will be saved as a cycle if cycles are enabled for that file. The user may specify the number of cycles to be maintained and their life space by setting the following attributes", "Cycle Limit", and "Cycle Life Span"},

wherein the determining whether the number of versioned files to maintain for the source file exceeds the specified maximum number is performed for each specified time period {See CROUSE, col. 15, lines 1-16, wherein this reads over "[w]hen a new version of a file is created the previous version will be saved as a cycle if cycles are enabled for that file. The user may specify the number of cycles to be maintained and their life space by setting the following attributes", "Cycle Limit", and "Cycle Life Span"}, and

wherein selecting the number of oldest versioned file names to purge to reduce the number of versioned file names in the retention index file is performed for each specified time period to reach the maximum number specified for the time period {See CROUSE, col. 21, lines 28-30, wherein this reads over "[t]he remote files that are eligible for removal or archiving and have waited longest since last access will be eliminated or archived first"}.

The combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the retention rule specifies a time limit on versioned files to be maintained for the source file (i.e. how long a versioned file of a source file may exist), and selecting the versioned files to purge comprises of determining whether the number of version file names exceeds a time limit (i.e. cycle life span), and selecting the versioned file names which exceed the time limit for purge (i.e. archiving or removing those files which have waited longest since last access or after a predetermined period of time has passed). Furthermore, the combination of inventions disclosed in CROUSE, HUG, GONOS, and KALER would disclose a method wherein the retention rule specifies a maximum number of versioned files to have for the source file (i.e. how many copies of a source file may exist), and selecting the versioned files to purge comprises of determining whether the number of version file names exceeds a maximum number (i.e. cycle limit), and selecting the oldest versioned file names for purge (i.e. archiving or removing those files which have waited longest since last access or after a predetermined period of time has passed). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the inventions suggested by CROUSE, HUG, GONOS, and KALER.

GONOS, CROUSE, HUG, and KALER are analogous art because they are from the same field of endeavor of data tracking and file versioning. One of ordinary skill in the art would have been motivated to do this modification so that older versioned files are systematically and routinely purged from the system.

26. **Claims 12, 19 and 34** are rejected under 35 U.S.C. 103(a) as being unpatentable over CROUSE, in view of HUG and GONOS, and in further view of Official Notice.

GONOS, CROUSE and HUG teach the limitations of claims 1, 2, 10-11, 13-15, 18, 20-21, 23-24, 32-33, AND 35 for the reasons stated above.

GONOS, CROUSE and HUG differ from the claimed invention in that they fail to disclose a method wherein a program executing in a kernel of an operating system (claims 12, 19, and 34).

27. **As per dependent claims 12, 19, and 34,** the claims are rejected under 35 U.S.C. 103(a) as being obvious in view of CROUSE, HUG, GONOS, and in further view of Official Notice.

It would have been obvious to one of ordinary skill in the art to have a program executed in a kernel of an operating system.

Response to Arguments

28. Applicant's arguments filed 24 August 2006 have been fully considered but they are not persuasive.

a. **Applicant's Arguments:**

i. 35 U.S.C. 103(a) based on Crouse

As per claims 1, 14, 20, and 23, Applicant asserts the argument that Crouse fails to disclose the claim limitation wherein "a retention rule is provided for the source file in response to the write request" (Amendment, page 14).

As per claims 1, 14, 20, and 23, Applicant asserts the argument that Crouse fails to disclose a method wherein "purging is initiated in response to a write request to a file for which a retention rule is provided and a versioned file is generated" (Amendment, page 16).

As per claims 9 and 31, Applicant asserts the argument that Crouse fails to disclose "multiple time periods and maximum numbers, i.e., multiple cycle limits and life spans, for one source file" (Amendment, page 19).

ii. 35 U.S.C. 103(a) based on Gonos

As per claims 1, 14, 20, and 23, Applicant asserts the argument that Gonos fails to disclose the claim limitation comprising "adding a generated versioned name to a retention index file" (Amendment, page 15).

As per claims 2, 15, 21, and 24, Applicant asserts the argument that Gonos fails to disclose the claim limitation comprising "deleting determined versioned file names of versioned files to purge from a retention file index" (Amendment, page 17).

As per claims 3, 16, 22, and 25, Applicant asserts the argument that Gonos fails to disclose the claim limitation comprising "applying a function to the source file name to determine the retention index file" (Amendment, page 18).

b. **Response to Arguments:**

i. 35 U.S.C. 103(a) based on Crouse

In response to applicant's arguments against Crouse individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986). One of ordinary skill in the art at the time the invention was made would be able to properly discern that the combined teachings of Crouse and Gonos would provide an invention wherein "a retention rule is provided for the source file in response to a write request." That is, Gonos provides a method wherein "[a] process is initiated when a determination is made that a particular dataset is going to be archived." Therefore, where the process makes a determination to archive a particular dataset (i.e. the request to write), the subsequent determination of whether a retention rule is provided for the source file would be done in response to the write request.

Furthermore, applicant is directed to column 21, lines 43-46 of Gonos which discloses a method wherein a determination is made "[i]f all selection criteria specified in the selection attributes are met by the file attributes and the file access information, that that remote file is eligible for archiving and/or removal." One of ordinary skill in the art would be able to correlate the determination of whether "[a] file is eligible for archiving" with the claimed write request.

Additionally, applicant argues that Crouse fails to disclose a method wherein "purging is initiated in response to a write request to a file for which a retention rule is provided and a versioned file is generated" (Amendment, page 16). In response to applicant's argument that the references fail to show certain features of applicant's invention, it is noted that the features upon which applicant relies (i.e. initiating a purge) are not recited in the aforementioned rejected claim(s). The claim limitations simply recite the processing of a retention index file to determine whether to purge versioned files, yet fail to positively recite a method wherein the purge is initiated. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993). It is noted that Crouse does not simply discloses purging remote files when disk space exceeds a threshold, but also discloses the selection of "remote files that are eligible for elimination or archiving in accordance with their hierarchically selectable archival file attributes" (C21:L24-32). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Crouse to have an invention wherein the file attributes are applied to the retention rule to determine whether to purge the files. Furthermore, as recited in claims 7 and 9 of the present invention, the use of file attributes such as timestamps provide an integral function of the claimed invention by allowing for the determination of whether a file has exceed a time limit.

For the aforementioned reasons, Applicant's arguments are not persuasive and the rejection of claims 1, 14, 20, and 23 are sustained under 35 U.S.C. 103(a).

As per claims 9 and 31, Applicant asserts the argument that Crouse fails to disclose "multiple time periods and maximum numbers, i.e., multiple cycle limits and life spans, for one source file" (Amendment, page 19). It is noted that Crouse discloses that "[t]he user may specify the number of cycles to be maintained and their life span by setting the following attributes" (C15:L1-5). It would have been obvious to one of ordinary skill in the art to discern that where the reference discloses "number of cycles" and the correlative "their life span" in the plural, that multiple time periods and maximum numbers for one source file would indeed be possible.

For the aforementioned reasons, Applicant's arguments are not persuasive and the rejection of claims 9 and 31 are sustained under 35 U.S.C. 103(a).

ii. 35 U.S.C. 103(a) based on Gonos

In response to applicant's arguments against Gonos individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

As per independent claims 1, 14, 20, and 23, one of ordinary skill in the art at the time the invention was made would be able to properly discern that the combined teachings of Gonos and Hug would provide a method of "adding a generated versioned name to a retention index file." Applicant is directed to Gonos, wherein the following is disclosed:

> "[A]n object database system that logically organized data into a series of objects. Each object is associated with a series of attributes, and each instance of an object is associated with a series of attribute values" and "[a]n archive file may store archive data, for example,

> from one or more relational database tables, object instances, or other types of data files" (C3:L30-47); and
>
> "[S]ome implementations may store information about more than one archive file in an archive index file" (C6:L55-60).

Furthermore, with the disclosed invention of Hug (i.e. "identification data preferable identifies the data and time the version was created, a version name associated with the version" (C6:L28-41)), one of ordinary skill in the art would been able to combine the teachings of Gonos and Hug such that "a version name associated with the version" may be added to "an archive index file." That is, while Gonos indeed does disclose the archival and deletion of database table rows, it would have been obvious to one of ordinary skill in the art that the combination of Hug and Crouse would allow for the rows to contain file version information. By applying the processor's ability to apply decision logic and add rows to the archival index, as found in Gonos, one of ordinary skill in the art could then combine said ability with the "set of hierarchically selectable archival attributes selectively assigned to each remote file," as found in Crouse, and "the version name," as found in Hug, to create a method wherein a version name is added to a retention index file.

As per independent claims 2, 15, 21, and 24, one of ordinary skill in the art at the time the invention was made would be able to properly discern that the combined teachings of Gonos and Hug would provide a method of "deleting the determined versioned names from the retention index file." Applicant is directed to Gonos, wherein the following is disclosed:

> "[T]he decision logic may be that row is deleted from separate storage after a predetermined number of days since the row was moved to separate storage or after the archiving process has been performed for the dataset a predetermined number of times since the row was moved to a separate storage" (C7:L43-50).

By applying the processor's ability to apply decision logic and delete rows from the archival index, as found in Gonos, one of ordinary skill in the art could then combine said ability with the "set of hierarchically selectable archival attributes selectively assigned to each remote file," as found in Crouse, and "the version name," as found in Hug, to create a method wherein a version name is deleted from a retention index file.

As per independent claims 3, 16, 22, and 25, one of ordinary skill in the art at the time the invention was made would be able to properly discern that the combined teachings of Gonos and Howard would provide a method of "applying a function to the source file name to determine the retention index file." Applicant is directed to Gonos, wherein the following is disclosed:

> "Each version entry contains a hash code or digest that to a very high probability uniquely identifies the contents of a corresponding version of the file" (C3:L31-38); and
>
> "Each version entry in each journal file contains a site indicator field indicating which of the journal files for the other site has a copy of the entry" (C3:L56-59).

Therefore, one of ordinary skill in the art at the time the invention was made would be able to integrate the invention disclosed by Howard to provide a method wherein hash codes, identifying different versions of the files, are computed from the contents of the file and thereafter are used to determine the retention index file. As best understood by the Office, the recitation of "determin[ing] the retention index file" would correlate with the method of tracking disclosed by Howard. That is, Howard discloses a method of tracking wherein "[e]ach version entry in each journal file contains a site indicator field indicating which of the journal files for the other sites has a copy of the entry" (C3:L55-58). While Howard provides a method wherein hash codes are computed from the contents of the file, it would have been obvious to one of ordinary skill in the art that the contents of the source file would necessarily and inherently contain the name of the source file. Thus, under Howard, one of ordinary skill in the art would have been able to compute a hash code from (i.e. apply a function to) the source file name to determine the retention index file.

For the aforementioned reasons, Applicant's arguments are not persuasive and the rejection of claims 1-3, 14-16, 20-22, and 23-25 are sustained under 35 U.S.C. 103(a).

Conclusion

29. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

30. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Paul Kim whose telephone number is (571) 272-2737. The examiner can normally be reached on M-F, 9am - 5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey Gaffin can be reached on (571) 272-4146. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Paul Kim
Patent Examiner, Art Unit 2161
TECH Center 2100

JEFFREY GAFFIN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100